maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

at the client, identifying an application program for update;

a

in response to a call to the server from the client, causing the server to download [the catalog] to the client the catalog for the identified application program and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

wherein substantially all processing, other than file transfers, in updating the application program is performed at the client.

4. (Amended) A method as claimed in claim 1 wherein the server downloads the catalog and components to the client for updating the application program on the client by way of hypertext transfer protocol.

a2

5. (Amended) A method as claimed in claim 1 wherein the server downloads the catalog and components to the client for updating the application program on the client by way of file transfer protocol.

a3

29. (Amended) A method as claimed in claim 1 wherein the network is an intranet and the client is a World Wide Web browser, and the call to the application program is made through a hypertext link on the browser directed to the catalog on the server[, the Web browser includes a launcher program, which is executed when the hypertext link to the catalog is selected, to update the application program and to execute the updated application program thereafter].

34

65. ~~60~~. (Amended) A programmed data processing client comprising:

an application program including components, each being provided with a version identification; and

a launcher program which, when a user selects the application program to execute the program:

causes a server to download a catalog to the client, the catalog specifying the components with the version identifications;

compares the version identifications of the components indicated in the downloaded catalog with version identifications of the components maintained on the client; and

updates the application program on the client by downloading from the server to the client the selected components for which the version identifications do not match and by replacing the selected components on the client.

A4

66. ~~61~~. (Amended) A program on a storage device providing instructions, for execution on a client, which:

when a user selects to execute an application program[,] on the client, the application program including components, each component having a version identification, cause a server to download a catalog of components with the version identifications;

compare the version identifications indicated in the catalog with the version identifications of the components maintained on the client; and

update the components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client.

70. ~~65~~. (Amended) A method of maintaining an application program in a client-server environment comprising:

A5

maintaining on a server the application program, the program including the components, each having a version identification, and maintaining a catalog of components with the version identifications;

35

A5 cont.

in response to a first call to the server from a client, causing the server to download the catalog to the client and the application program;

maintaining on the client the application program and information in the downloaded catalog including a list of the components with the version [idendifications] identifications;

in response to a subsequent call to the server from the client, causing the server to download a second catalog including the latest version identifications of the components and of any new additional components on the server;

comparing in the client the latest version identifications of the components in the second catalog with the version identifications of the components maintained on the client;

updating the application program on the client by downloading from the server to the client the selected components for which the version identifications do not match; and

executing the updated application program on the client.

Please add the following new claims:

30. ~~66~~. A method as claimed in claim 29 wherein the Web browser includes a launcher program, the launcher program being executed when the hypertext link to the catalog is selected on the browser, to update the application program and to execute the updated application program thereafter.

a6

31. ~~67~~. A method as claimed in claim ~~66~~ 30 wherein the launcher program is a plug-in module to cooperatively run with the browser.

32. ~~68~~. A method as claimed in claim ~~66~~ 30 wherein the launcher program is a helper application to cooperatively run with the browser.

33. ~~69~~. A method as claimed in claim 29 further comprising specifying in the catalog a procedure to install an icon on the client which enables a user to subsequently execute the application program without accessing the browser.

36

34. ~~70.~~ A method as claimed in claim 29 further comprising specifying in the catalog a procedure to determine whether a launcher program is present on the client, and if the launcher program is not present, retrieving from the server the launcher program and installing the program into the Web browser.

71. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client wherein the catalog maintained on the server includes network addresses of further servers from which the components can be retrieved.

72. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client wherein the catalog includes procedures for executing supplemental programs on the client prior to executing the updated application program.

73. A method as claimed in claim 72 wherein the supplemental programs are executed following the execution of the application program.

74. A method as claimed in claim 72 wherein the supplemental programs include a virus scanning program.

a6

75. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising specifying a maximum wait-time interval to limit any delay associated with updating the application program, and in a catalog maintained on the client specifying a list of further servers on the network, each including a copy of the catalog, such that when the server fails to download the catalog within the maximum wait-time interval, the client cancels the download and routes the call to one of the further servers to engage a new download and so on until the catalog has been downloaded within the specified maximum wait-time interval.

76. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

a4

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising:

specifying in the catalog a cryptographic digest for each component to ensure authenticity and integrity of the component;

updating the application program on the client by downloading from the server to the client and replacing the selected components for which the cryptographic digests do not match; and

computing the cryptographic digests on the client to ensure that the downloaded components are authentic and that the components have not been corrupted during transmission.

77. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising specifying in the catalog any component of the application program which are no longer needed to execute the program; and

in response to the call to the application program from the client, updating the application program by downloading the catalog from the server to the client and deleting the selected components on the client that are no longer needed prior to executing the updated program.

a6

78. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client the catalog for the identified application program and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising specifying a time interval in the catalog, and the application program is updated only on a first time the application is run in the specified time interval.

79. A method as claimed in claim 78 wherein the time interval is specified by a user from the client.

80. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

a4

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising specifying a fixed time interval in the catalog at which the application program is updated regardless of whether the call to the server is made from the client.

81. A method as claimed in claim 80 wherein the application program is updated at the fixed time interval only as necessary to maintain the application program current without executing the application.

82. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client the catalog for the identified application program and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

wherein the application program on the client is automatically updated on the client each time the client is booted up.

83. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

a4

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

wherein the components include executable codes, library files, parameter files, and data files of the application program.

84. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

wherein the network is the Internet, the server is an Internet server, and the client is a World Wide Web browser, and the call to the server is made through a hypertext link on the browser directed to the catalog on the server.

a4 85. A method as claimed in claim 84 wherein the Web browser includes a launcher program, the launcher program being executed when the hypertext link to the catalog is selected on the browser, to update the application program and to execute the updated application program thereafter.

86. A method as claimed in claim 85 wherein the launcher program is a plug-in module to cooperatively run with the browser.

87. A method as claimed in claim 85 wherein the launcher program is a helper application to cooperatively run with the browser.

88. A method as claimed in claim 85 further comprising specifying in the catalog a procedure to install an icon on the client which enables a user to subsequently execute the application program without accessing the browser.

89. A method as claimed in claim 85 further comprising specifying in the catalog a procedure to determine whether a launcher program is present on the client, and if the launcher program is not present, retrieving from the server the launcher program and installing the program into the Web browser.

90. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

wherein the network is an intranet and the client is a World Wide Web browser, and the call to the application program is made through a hypertext link on the browser directed to the catalog on the server.

91. A method as claimed in claim 90 wherein the Web browser includes a launcher program, which is executed when the hypertext link to the catalog is selected, to update the application program and to execute the updated application program thereafter.

92. A method as claimed in claim 91 wherein the launcher program is a plug-in module to cooperatively run with the browser.

93. A method as claimed in claim 91 wherein the launcher program is a helper application to cooperatively run with the browser.

94. A method as claimed in claim 91 further comprising specifying in the catalog a procedure to install an icon on the client which enables a user to subsequently execute the application program without accessing the browser.

95. A method as claimed in claim 91 further comprising specifying in the catalog a procedure to determine whether a launcher program is present on the client, and if the launcher program is not present, retrieving from the server the launcher program and installing the program into the Web browser.

96. A method of maintaining application program components on a network comprising:

maintaining on a server the application program, the program including components, each having a version identification, and maintaining a catalog of components with the version identifications;

maintaining the application program on a client;

in response to a call to the server from the client, causing the server to download the catalog to the client and, in the client, comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client;

updating the application program components on the client by downloading from the server to the client the selected components for which the version identifications do not match and replacing the selected components on the client; and

executing the updated application program on the client;

the method further comprising specifying in the catalog a procedure to delete the components immediately after executing the updated application to free up disk space on the client.

## REMARKS

Applicant thanks Examiner Smithers for a helpful telephone interview with the undersigned on November 20. In that interview the distinctions between the independent claims and the Cole reference were discussed.

Applicant notes with appreciation the allowance of claims 6, 10-20, 22-30, 36, 41-51, 53-59 and 63-64. Those claims remain dependent on the amended base claims. In addition, claims 6, 10-20 and 22-30 have been rewritten into independent form as claims 71-90 and 96 without the limitations being added by this amendment to the base claim 1. Note that claim 90 parallels amended claim 29 as noted below and claims 91-95 parallel new claims 66-70.